

Asterisk and intercepting unassigned numbers

In the telecom world, we often have to deal with blocks of Direct-in-Dial (DID) numbers that we rent but have not assigned to users yet. In some telephone systems, all unassigned numbers ring to the console attendant or local operator if they are dialed.

This can be a major problem if telemarketers use a robo-dialer and just start calling all numbers in a particular prefix one after another.

For many decades, the central offices for the telephone companies provided a feature called “intercept” where unassigned number rang to a recording to alert callers that they had misdialed.

The recording was often preceded by a SIT or Special Information Tone to alert callers that something was amiss. Over the years, telemarketing equipment manufactures took advantage of this tone to allow their equipment to mark a number as not callable and remove it from their database. This made the telemarketer dialer more efficient as it didn’t waste time calling vacant numbers.

Imagine your paid receptionist/operator not getting anything done if they get dozens if not hundreds of calls one right after the other from an unwanted telemarketer?

We need to deal with unassigned numbers and in the process maybe reduce the number of robo-calls from telemarketers and political parties.

This also provides an internal feature that alerts your employees that they have misdialed a number and reached one that does not exist.

While this simple context was written for a VoIP class, it is a real world example of how to do this.

```
[intercept]
exten => _XXXX,1,Answer()
exten => _XXXX,n,Zapateller()
exten => _XXXX,n,Playback(the-number-u-dialed)
exten => _XXXX,n,SayDigits(${EXTEN})
exten => _XXXX,n,Playback(no-longer-in-service)
exten => _XXXX,n,Playback(vm-goodbye)
exten => _XXXX,n,Hangup()
```

Normally we would just put an **Include => intercept** in our default context for our pbx but in our classroom situation where we use a `_XXXX` pattern match to dial other systems, we need to point a wild card extension to this context that does not match `_XXXX`

The following extension in your dialplan above the `_XXXX` extension will redirect to this context for extensions that are in your number plan but not assigned.

```
exten => _NNXX,1,Gosub(intercept,${EXTEN},1)
```

Change the NN to the first two digits of your extensions in your system.

What this does is match a four digit number starting with your two digit extension prefix, send it to the intercept context along with the 4 digits you dialed and point it to priority one of the intercept context.

Here is a slightly different version of [intercept]

```
[intercept]
exten => _XXXX,1,Answer()
exten => _XXXX,n,Zapateller()
exten => _XXXX,n,Playback(discon-or-out-of-service)
exten => _XXXX,n,Wait(1)
exten => _XXXX,n,SayDigits(${EXTEN})
exten => _XXXX,n,Hangup()
```

I found a recording in Asterisk that said the number was disconnected or out of service and in this version, it plays the SIT tone, then the recording, then plays back the digits dialed and hangs up.

This one sounds very much like the old central office recordings.

However, in them, the numbers announced at the end were for diagnostic purposes and told the technician from which central office the intercept message originated. Sometimes they even announced the name of the central office exchange.